Optimization and Realization of Boson Sampling for True Random Number Generation using the Xanadu X8

Joshua Ange[a] and Mitchell A. Thornton[a]

[a]Darwin Deason Institute for Cyber Security, Southern Methodist University,
6425 Boaz Lane Dallas, TX 75205, USA

ABSTRACT

Random number generators (RNGs) are critical for problems involving security and cryptography. Much work has been done with true random number generators (TRNGs) that utilize quantum mechanical phenomena as sources of entropy to generate numbers according to some distribution without any underlying deterministic component. While qubit-based quantum computers can serve as high-quality sources of randomness, their effectiveness is limited by constraints due to their physical construction (e.g. not being feasible at room temperatures) and by the number of qubits that are necessary for fine-grained distributions (the number of bins scale as 2^n for a n-qubit implementation). Continuous variable photonic quantum computers offer promising scalability, both from their experimental realization and from the use of linear optical components and photon number resolving detectors to generate varied distributions. We demonstrate that boson-sampling-based approaches realized with photonic quantum computers like the Xanadu X8 machine can be used to generate random numbers in diffuse, hard-to-predict distributions and they can be optimized for transformation to a uniform distribution. Specifically, boson sampling can act as a high-quality source of entropy and we introduce a strategy for scaling our TRNG with further experimental development (both in terms of the number of qumodes available and improvements for photonic components). We compare our results to the performance of qubit-based TRNGs and discuss experimental realizations.

Keywords: Random number generation, true random number generation, quantum random number generation, near-term quantum computing

1. INTRODUCTION

Random number generators (RNGs) are crucial for problems involving security and cryptography. The prevalence of communication architectures, networking, and data encryption/decryption demands the ability to generate high-quality random numbers (i.e. with reliability and at high speed).[1] Classical schemes for RNGs include pseudorandom number generators (PRNGs) that deterministically generate a seemingly random sequence of numbers from an initial seed. For true random number generators (TRNGs), a "weakly random source" of entropy is employed that makes use of a source of randomness that arises in nature. A natural choice for this phenomena is that which is quantum in nature, leading to the notion of quantum random number generators (QRNGs).[2,3]

A central concept in realizing QRNGs is to exploit the superposition of a qubit $|\psi\rangle$ that is in equal superposition between two states $|0\rangle$ and $|1\rangle$ such that $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. By measuring the qubit with respect to the computational basis, one has an equal probability of observing either state. This measurement thus serves as a TRNG that effectively performs a Bernoulli trial with an equal likelihood of success/failure. Practical difficulties arise in implementations that exploit this fundamental idea due to hardware biases and other noise sources that can pollute the desired distribution and affect the number of possible outputs per qubit. As any n-qubit architecture can only support a maximum of 2^n possible output states, we hypothesize that higher-dimensioned qudit-based architectures and continuous-variable qumodes may be more effective for generating distributions without requiring an exorbitant number of quantum information units.[4] Discrete qudit-based architectures have been considered[5] and we explore a continuous-variable approach in this work.

Send correspondence to JA: jwange@smu.edu. Further author information: MAT: mitch@smu.edu

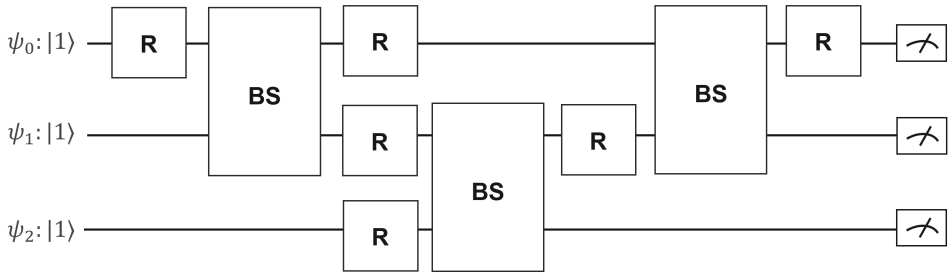

Figure 1. Boson sampling architecture for 3 qumodes, as expressed with photonic components. 'R' refers to phase space rotation gates and 'BS' refers to beamsplitters.

Continuous-variable quantum computing is a paradigm distinct from discrete-variable qubit-based computing. Rather than being expressed in terms of a discrete basis, a continuous-variable qumode $|\phi\rangle = \int dx\, \phi(x)\, |x\rangle$ is used, where the $|x\rangle$ states are eigenstates of the $\hat{x} = \sqrt{\frac{\hbar}{2}}(\hat{a} + \hat{a}^\dagger)$ quadrature (where $\hat{a}$ is the typical annihilation operator). Equivalently, they can also be expressed in terms of the momentum quadrature $\hat{p} = -i \cdot \sqrt{\frac{\hbar}{2}}(\hat{a} - \hat{a}^\dagger)$. As implied by the form of these quadratures involving annihilation and creation operators, the continuous-variable formulation of quantum phenomena is a particularly useful representation for photonic quantum computing systems.[6] Furthermore, there is reason to believe that continuous-variable computation may be beneficial for practical TRNG development due to room-temperature realizations[7] and a continuum of outputs that potentially enable higher-resolution final distributions (for example, compared to the output of 2 possible states per qubit).[4]

Xanadu's X8 photonic quantum computer supports qumode-based computing[8] and their Strawberry Fields and PennyLane tools support continuous-variable simulation capabilities. In this work, we use Xanadu's tools to create a proof-of-concept QRNG based on boson sampling that is designed to yield a uniform distribution. Additionally, we simulate hardware noise through photon loss and imprecise component parameters (and therefore also proxy channel crosstalk, the other main source of experimental noise).

2. CIRCUIT ARCHITECTURE AND PROCEDURE

2.1 Boson Sampling as an Entropy Source

The Xanadu X8 machine exclusively uses photon number resolving (PNR) detectors and there is not any direct homodyne measurement possible, thus the ideal approach of direct measurements of position and momentum quadratures is not possible.[6,8] Boson sampling is considered a powerful example of an "intermediate quantum computer," designed to experimentally implement a computation that is intractable classically. In particular, boson sampling involves the evolution of single photons through a linear network described by a single unitary matrix, which is implemented solely from beam splitter and phase shifter components. These gates mix the input modes coherently and redistribute the photons across the output modes with a probability that is, in general, difficult to compute, as it involves the permanents of submatrices of the unitary tr ansformation.[9] Boson sampling, and particularly Gaussian boson sampling, involves only passive optical components resulting in a continuous improvement in detection efficiencies an d re ductions in lo ss ar e po ssible,[10] in pr inciple, fo r a large number of photons.[11]

Boson sampling has been proposed as a source of entropy in previous works,[12,13] but often the full utilization of the diffuse output probability distribution is not made (for example, only measuring modes in the Fock basis as either zero or non-zero, and not considering various mode measurements). In this work, we use the full diffuse o utput p robability o f b oson s ampling a nd e xtract t he m aximum p ossible i nformation f or randomness. Additionally, as opposed to discrete-variable qubit-based architectures,[14] we show that sources of noise like photon loss are not prohibitive with respect to random number generation (and in some cases, higher levels of photon loss actually *increases* the resemblance to a uniform distribution).

	0	**1**	**2**	**3**
0	0	1	2	3
1	4	5	6	7
2	8	9	A	B
3	C	D	E	F

Figure 2. Table to generate radix-$(3+1)^2$ number from Fock measurements.

2.2 Architecture and Transformation

The overall architecture for boson sampling can be equivalently expressed with a unitary matrix or with a series of photonic components, as shown in Fig. 1. Our general architecture for a QRNG with $2N$ available qumodes is structured as follows:

1. Initialize all $2N$ qumodes to single photon $|1\rangle$ Fock states
2. Evolve the first N qumodes with a boson sampling network represented by $N \times N$ unitary matrix U_1, and evolve the second N qumodes with a boson sampling network represented by $N \times N$ unitary matrix U_2
3. Measure all $2N$ qumodes in the Fock basis
4. Use Fock measurements to generate radix-$(N+1)^2$ number
5. "Extraction," or post-processing to transform the generated values into a uniform distribution

For the majority of this paper, we focus on the architecture for $2 \times 3 = 6$ qumodes to generate random numbers. We also found success with $2 \times 2 = 4$ qumodes, but the benefits of this smaller QRNG as compared to a qubit-based implementation is minimal. Furthermore, in principle, the procedure would work utilizing all $2 \times 4 = 8$ qumodes of the X8 (or more, given further availability), but the optimization of the unitary matrix becomes numerically difficult.

Note that, to generate a radix-$(N+1)^2$ number in step 4, one constructs a table such as that in Fig. 2. Given two Fock measurements i and j (one from either set of N qumodes), a radix-$(N+1)^2$ digit is generated by selecting the element in the $(i+1)$th column and $(j+1)$th row. For example, for $N = 3$, if the measurements from the first N qumodes are $2, 0, 1$ and the measurements from the second N qumodes are $3, 0, 0$, then the resultant radix-16 number is E01 (or 3585 in the more familiar base-10).

The extraction process in step 5 involves taking the cumulative distribution function (CDF) of the radix-$(N+1)^2$ number distribution and performing a transformation such that the new distribution approximates a uniform distribution over $[0, 1]$. This distribution has $\binom{2N+1}{N-1}^2$ non-identity points that can be collected into a chosen number of 'bins' for a final distribution. Note that while, in principle, this would lead one to believe that one could construct a uniform distribution of $\binom{2N+1}{N-1}^2$ bins, the conservation of photons means there will invariably be smaller probabilities for numbers at the extremes (e.g. near 000_{16} or FFF_{16}), so that distribution will never be appropriately uniform. In practice, the difficulty fo r th is pr ocedure co mes by op timizing the parameters of the boson sampling architecture for a desired number of output bins.

3. OPTIMIZATION AND SIMULATION

3.1 Parameterization

As mentioned in Sec 2.2, the true difficulty in volved in th e bo son sa mpling ar chitecture co mes in optimizing its parameters in order to approach a uniform distribution. In the case of six qumodes, we use two 3×3

unitary matrices that each evolve the states of three qumodes with a boson sampling network. We follow the parameterization[15] where any 3×3 unitary matrix U can be expressed as

$$U = Q \begin{bmatrix} e^{i\alpha_1} \cos\chi & ie^{i\alpha_2} \cos\mu \sin\chi & ie^{i\alpha_3} \sin\mu \sin\chi \\ ie^{i\alpha_1} \sin\chi & e^{i\alpha_2} \cos\mu \cos\chi & e^{i\alpha_3} \sin\mu \cos\chi \\ 0 & e^{i\beta_2} \sin\mu & -e^{i(\beta_2 - \alpha_1 + \alpha_3)} \cos\mu \end{bmatrix} Q^T$$

$$Q = \begin{bmatrix} \cos\phi\cos\theta\cos\varphi + \sin\phi\sin\varphi & -\cos\phi\cos\theta\sin\varphi + \sin\phi\cos\varphi & \sin\theta\cos\phi \\ -\sin\phi\cos\theta\cos\varphi + \cos\phi\sin\varphi & \sin\phi\cos\theta\sin\varphi + \cos\phi\cos\varphi & -\sin\phi\sin\theta \\ -\sin\theta\cos\varphi & \sin\theta\sin\varphi & \cos\theta \end{bmatrix}$$

where $\phi \in (-\pi, \pi]$, $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, $\varphi \in [0, \pi)$, $\chi \in [-\frac{\pi}{4}, \frac{\pi}{4}]$, $\mu \in [0, \frac{\pi}{2}]$, and $\alpha_1, \alpha_2, \alpha_3, \beta_2 \in [0, \pi]$. These nine variables are those across which we attempt to minimize the distance between our generated distribution and true uniformity (but in principle could be used to minimize distance to *any* desired distribution). This parameterization is used as a matter of convenience, but does not actually reduce (nor increase) the number of free parameters corresponding to the boson sampling system (see Fig. 1).

For a 4×4 matrix, the analogous parameterization would require minimization across at least 16 variables. But for all of the minimization techniques and cost functions attempted, none were able to minimize the high-dimensional space such that the final distribution resembled the uniform distribution to a similar degree as the 3×3 version. This is another reason we focus on the architecture for six qumodes rather than utilizing the full count of eight made available by the X8.

3.2 Optimization Techniques

When it comes to minimizing the distance between the generated distribution and an ideal uniform distribution, there is no obvious metric. This is especially the case when 'overbinning,' as discussed below, where the height of multiple entries in the histogram may be zero and so metrics like mean-squared error (MSE) are not always best to optimize for. Thus, taking p and q to be our generated distribution and that of an ideal uniform output, we consider the 1-Wasserstein distance

$$\mathrm{WS}(p||q) = \left(\frac{1}{n} \sum_{i=1}^{n} |p(x_i) - q(x_i)| \right),$$

the Kullback–Leibler divergence

$$\mathrm{KL}(p||q) = \sum_{i=1}^{n} p(x_i) \log\left(\frac{p(x_i)}{q(x_i)} \right),$$

and the Jensen–Shannon divergence

$$\mathrm{JS}(p||q) = \sqrt{\frac{\mathrm{KL}(p||m) + \mathrm{KL}(q||m)}{2}}$$

(where m is the pointwise mean between p and q) that all intend to measure the similarity between their argument distributions in order to optimize the parameters of the boson sampling network.

We explored numerous optimization techniques. Markov Chain Monte Carlo (MCMC) fails to converge in any tractable timescale, likely due to the high correlation between any individual parameter of the unitary matrix and the final distribution. Newton-Raphson was quick in its optimization, but failed to achieve uniform distributions even for simple cases. Nelder-Meade was the most successful optimization technique, but as it is prone to falling into local minima, there may be parameter values that are closer to the true optimal values than those presented here.

An important notion for optimizing boson sampling architecture is that of 'overbinning' and 'underbinning.' Given a desired final distribution of 25 bins, one may optimize the boson sampling parameters first to 10 bins (in step 5), and then perform a secondary CDF transformation to the desired 25 bins ('underbinning'). Or

one may optimize first to 1000 bins, and then perform a secondary CDF transformation to the desired 25 bins ('overbinning'). The benefit of overbinning may seem evident, as one could, in principle, take an overbinned distribution and then choose a desired distribution at a later date, but in practice this underperforms 'proper' binning (i.e. neither overbinning nor underbinning). Where overbinning shows significant utility is with respect to increased hardware noise, as shown in Sec 4.1.

3.3 Results

Because Nelder-Meade is prone to falling into local minima, we ran the optimization procedure 100 times for each of the three distance metrics. The final distance metrics to the uniform distribution can be seen in Table 1.

	Distance Type	*Native 10 bins*	*Native 25 bins*	*Native 100 bins*	*Native 1000 bins*
To 10 bins	KL	8.360×10^{-7}	3.954×10^{-5}	5.501×10^{-5}	1.761×10^{-5}
	JS	2.454×10^{-7}	1.188×10^{-4}	3.215×10^{-5}	1.227×10^{-5}
	WS	2.195×10^{-6}	3.293×10^{-4}	3.372×10^{-5}	1.905×10^{-4}
To 25 bins	KL	1.548×10^{-4}	1.058×10^{-5}	2.830×10^{-5}	3.210×10^{-5}
	JS	1.347×10^{-4}	8.024×10^{-6}	3.559×10^{-5}	2.608×10^{-5}
	WS	2.740×10^{-4}	2.197×10^{-5}	4.473×10^{-5}	1.055×10^{-4}
To 100 bins	KL	1.765×10^{-4}	8.669×10^{-5}	2.800×10^{-5}	3.042×10^{-5}
	JS	1.153×10^{-4}	9.196×10^{-5}	3.422×10^{-5}	3.003×10^{-5}
	WS	1.472×10^{-4}	1.114×10^{-4}	3.566×10^{-5}	8.229×10^{-5}
To 1000 bins	KL	2.359×10^{-5}	1.452×10^{-5}	1.114×10^{-5}	1.014×10^{-5}
	JS	1.799×10^{-5}	1.669×10^{-5}	1.103×10^{-5}	1.015×10^{-5}
	WS	2.112×10^{-5}	1.710×10^{-5}	1.172×10^{-5}	1.484×10^{-5}

Table 1. Best MSE of parameter optimization to different distance metrics across Nelder-Meade runs. 'Native' bins refers to the number of bins in the distribution that is optimized in step 5 of the procedure and the bin count on the left hand side refers to a secondary transformation. Thus entries along the diagonal are *properly* binned, those in the upper right half are *over*binned, and those in the lower left half are *under*binned.

It is easy to see that the best performing metrics for the goal of 10 and 25 uniform bins is the Jensen-Shannon divergence, with MSE of 2.454×10^{-7} and 8.024×10^{-6}, respectively. The raw outputs (as arising from step 4) and post-processed outputs are observed in Figs 3 and 4, respectively. Additionally, it is clear that neither overbinning nor underbinning are helpful, in this case.

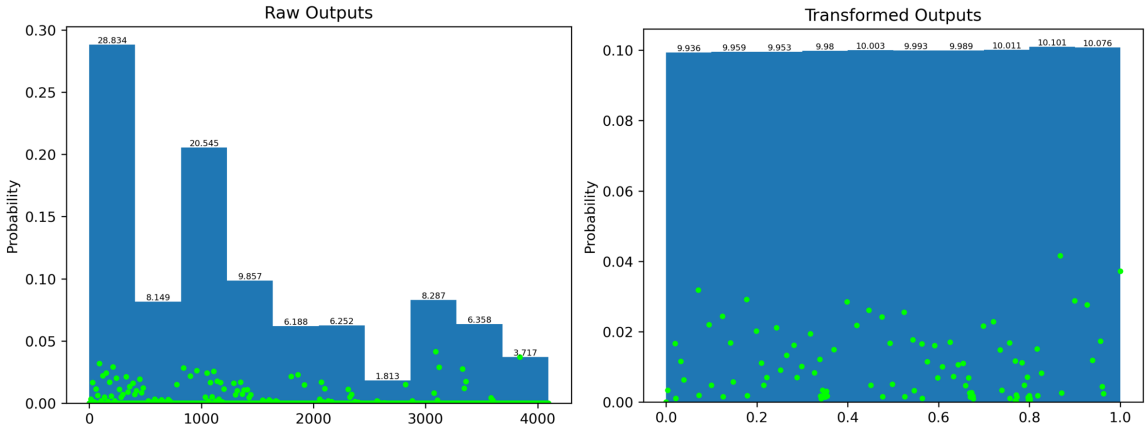

Figure 3. Results for 10-bin distribution from step 4 (left) and step 5 (right).

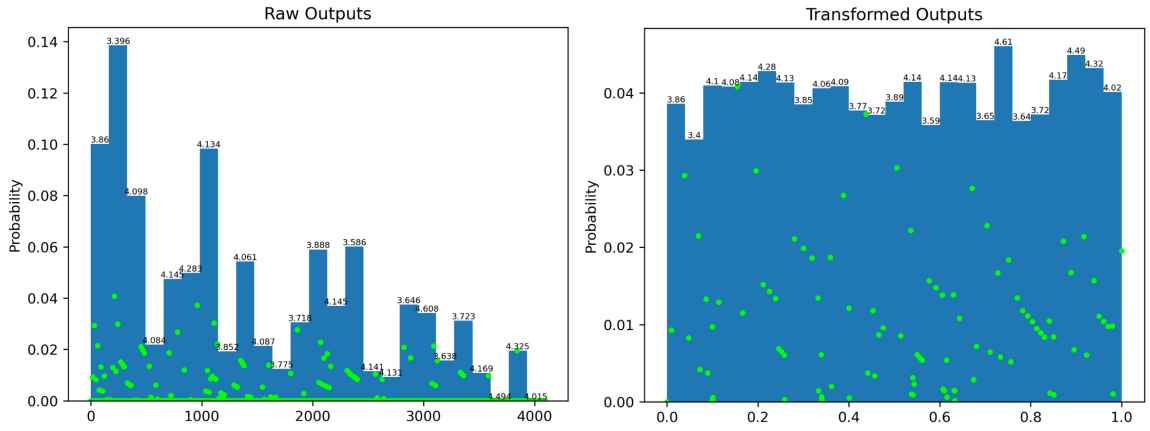

Figure 4. Results for 25-bin distribution from step 4 (left) and step 5 (right).

4. EXPERIMENTAL NOISE

4.1 Noise and Interference

The biggest sources of hardware noise for the Xanadu X8 are photon loss, entanglement strength, and susceptibility to crosstalk between qumodes.[8] Thus modeling such inefficiencies in real hardware is key to recognizing the success of this QRNG methodology in practice.

4.2 Results

We introduce photon loss into the circuit prior to our parameter optimization, and observe how the resemblance to uniformity changes with varying photon loss rates. As seen in Fig. 5, while photon loss always results in a larger minimum distance between our resultant distribution and uniformity, there is not a strict decrease in performance with increasing rates of photon loss. In particular, as the loss rate increases, the number of possible radix-$(N+1)^2$ numbers generated increases (because conservation of photon count is not such a strict constraint), and thus in principle allows for more performant parameter values.

Additionally, this is where we see overbinning has a positive effect on the final distributions. This is likely because the increased number of raw outputs are more aptly able to be 'picked out' by the finer bin number, and so the final CDF more closely brings us to uniformity. Thus, our QRNG architecture is fairly resistant to increasing rates of photon loss, so long as overbinning is employed.

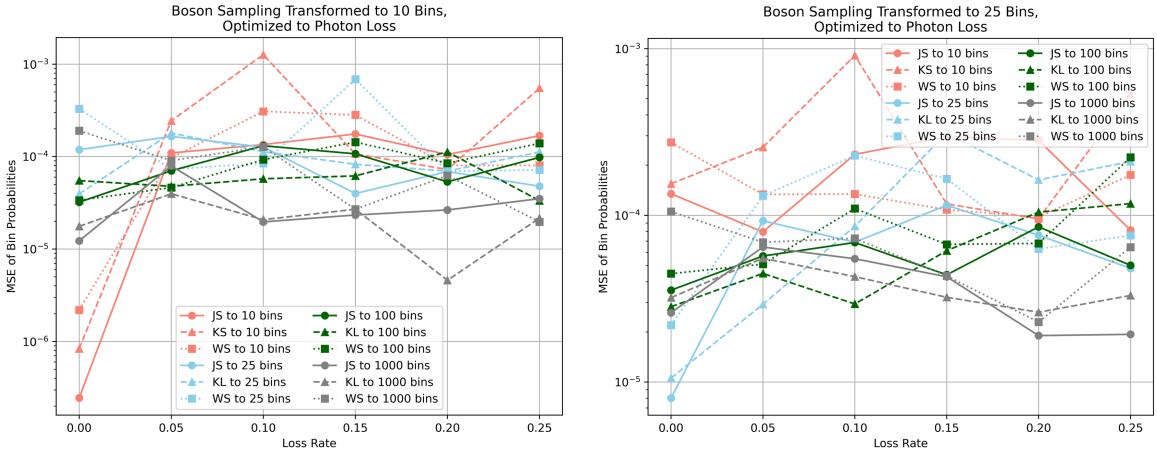

Figure 5. Results for QRNG architecture with photon loss as transformed to 10 bins (left) and 25 bins (right) from various native bin counts.

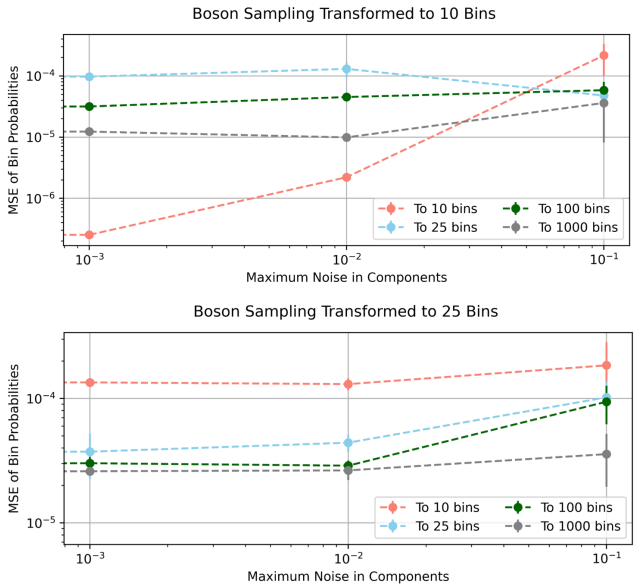

Figure 6. Average results for QRNG architecture with imprecision in photonic components.

The other main source of hardware noise comes from entanglement strength and crosstalk between the qumodes. Because, in our boson sampling architecture, all the qumodes are interacting with one another, this can be modeled in simulation as an imprecision in the optimized parameters. In particular, taking the Jensen-Shannon optimized values, we introduce varying levels of noise in the simulated component parameters, and record the influence on the final MSE of bin probabilities to a uniform distribution. This can be seen in Fig. 6

Again, here we see that overbinning is able to preserve relatively low MSE when introducing larger noise values in our parameters. One should also note that this is based on the idea that the CDF used in step 5 for the final transformation is built by recording the output of the architecture over time (rather than being known or calculated a priori). This is because rates such as photon loss or crosstalk between components are difficult, if not impossible, to know in advance. But given a satisfactory amount of time to generate such a CDF, this architecture is relatively secure against these sources of hardware noise.

5. CONCLUSION

In this work, we have demonstrated the viability of boson sampling as a source of entropy for TRNG using the Xanadu X8 photonic quantum computer. Photonic architectures not only provide a scalable and experimentally feasible alternative to qubit-based QRNGs, but also mitigate common noise sources. By levaraging continuous-variable quantum computing and PNR detection, we have outlined an architecture that generates high-quality random numbers, even in the presence of such hardware noise like photon loss and crosstalk between modes.

Future work could explore extending this methodology to larger qumode counts (i.e. either directly optimizing gate parameters or finding an optimal parameterization for unitary matrices) and optimizing for specific randomness applications such as cryptographic key generation and secure communications. Additionally, further work must be done to model different sources of experimental noise.

ACKNOWLEDGMENTS

We gratefully acknowledge the use of Xanadu's Quantum Cloud resources for this work. All views are those of the authors, and do not reflect the official policy or position of Xanadu or their respective teams.

REFERENCES

[1] Haylock, B., Peace, D., Lenzini, F., Weedbrook, C., and Lobino, M., "Multiplexed quantum random number generation," *Quantum* **3**, 141 (May 2019).

[2] Herrero-Collantes, M. and Garcia-Escartin, J. C., "Quantum random number generators," *Reviews of Modern Physics* **89** (Feb. 2017).

[3] Herrero-Collantes, M. and Garcia-Escartin, J. C., "Quantum random number generators," *Rev. Mod. Phys.* **89**, 015004 (Feb 2017).

[4] Michel, T., Haw, J. Y., Marangon, D. G., Thearle, O., Vallone, G., Villoresi, P., Lam, P. K., and Assad, S. M., "Real-time source-independent quantum random-number generator with squeezed states," *Phys. Rev. Appl.* **12**, 034017 (Sep 2019).

[5] Smith, K. N., MacFarlane, D. L., and Thornton, M. A., "A quantum photonic trng based on quaternary logic," in [*2020 IEEE 50th International Symposium on Multiple-Valued Logic (ISMVL)*], 164–169, IEEE (2020).

[6] Andersen, U. L., Leuchs, G., and Silberhorn, C., "Continuous variable quantum information processing," (2010).

[7] Na, N., Hsu, C.-Y., Chen, E., and Soref, R., "Room-temperature photonic quantum computing in integrated silicon photonics with germanium–silicon single-photon avalanche diodes," *APL Quantum* **1** (Sept. 2024).

[8] Ranjan, A., Patel, T., Gandhi, H., Silver, D., Cutler, W., and Tiwari, D., "Experimental evaluation of xanadu x8 photonic quantum computer: Error measurement, characterization and implications," in [*Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*], *SC '23*, Association for Computing Machinery, New York, NY, USA (2023).

[9] Lund, A., Laing, A., Rahimi-Keshari, S., Rudolph, T., O'Brien, J., and Ralph, T., "Boson sampling from a gaussian state," *Physical Review Letters* **113** (Sept. 2014).

[10] Tillmann, M., Dakić, B., Heilmann, R., Nolte, S., Szameit, A., and Walther, P., "Experimental boson sampling," *Nature Photonics* **7**, 540–544 (Jul 2013).

[11] Hamilton, C. S., Kruse, R., Sansoni, L., Barkhofen, S., Silberhorn, C., and Jex, I., "Gaussian boson sampling," *Physical Review Letters* **119** (Oct. 2017).

[12] Ahmad, I., "Quantum random number generation using boson sampling: Harnessing unbiased sequences from penceval and strawberry fields platforms," **9**, 114–123 (05 2024).

[13] Shi, J., Zhao, T., Wang, Y., Yu, C., Lu, Y., Shi, R., Zhang, S., and Wu, J., "An unbiased quantum random number generator based on boson sampling," (2022).

[14] Li, Y., Fei, Y., Wang, W., Meng, X., Wang, H., Duan, Q., and Ma, Z., "Quantum random number generator using a cloud superconducting quantum computer based on source-independent protocol," *Scientific Reports* **11**, 23873 (Dec 2021).

[15] Gil, J. J., "Parametrization of 3×3 unitary matrices based on polarization algebra," *The European Physical Journal Plus* **133** (May 2018).